Computers & Education 57 (2011) 1686–1697

Contents lists available at ScienceDirect

Computers & Education

journal homepage: www.elsevier.com/locate/compedu

Developing a remote laboratory for engineering education

E. Fabregas [a], G. Farias [a], S. Dormido-Canto [a,*], S. Dormido [a], F. Esquembre [b]

[a] Dpto. de Informática y Automática, Universidad Nacional de Educación a Distancia, ETSI Ingeniería de Informática, Juan del Rosal, 16 28040, Madrid, Spain
[b] Dpto. de Matemáticas, Universidad de Murcia, Murcia, Spain

ARTICLE INFO

Article history:
Received 25 June 2010
Received in revised form
14 February 2011
Accepted 15 February 2011

Keywords:
Engineering education
Remote laboratories
Simulink
EJS

ABSTRACT

New information technologies provide great opportunities for education. One such opportunity is the use of remote control laboratories for teaching students about control systems. This paper describes the creation of interactive remote laboratories (RLs). Two main software tools are used: Simulink and Easy Java Simulations (EJS). The first is a widely used tool in the control community, whereas the second is an authoring tool designed to build interactive applications in Java without special programming skills. The RLs created by this approach give students the opportunity to perform experiments with real equipment from any location, at any time, and at their own pace. The paper ends with an evaluation of this approach according to students' criteria and academic results.

1. Introduction

Control engineering education must adapt to the opportunities that information and communication technologies provide (Dormido, 2004; Heck, 1999). In this context, traditional laboratories can benefit from the Internet, which can be used by students for remote access to laboratory equipments (or plants). Remote operation of real plants is commonly known as remote laboratories (RLs) and can be incorporated into control engineering courses in order to avoid typical constraints of traditional laboratories, such as scheduling, cost of equipment and location. Although, simulations or virtual laboratories can be also used to overcome the disadvantages of traditional laboratories, any simulation is simply a model of a physical process, which is just an approximation that can not reproduce every aspect of the real phenomenon. So, the use of remote laboratories can be considered as an intermediate activity between simulations and traditional laboratories.

Recently, the control community has made many advances in implementing remote laboratories (Dormido, Vargas, Duro, Sánchez, & Dormido-Canto, 2008; Gomes & Bogosyan, 2008; Gomes et al., 2007; Jara et al., 2009; Lazar & Carari, 2008). However, much work remains to improve these learning resources from a pedagogical point of view. Among others, visualisation and interactivity are two interesting features that can be considered as criteria for remote laboratories used for pedagogical purposes (Dormido, 2004; Sánchez, Dormido, & Esquembre, 2005).

In control engineering, typical analysis of system response is performed on various characteristics of output signals (such as waveform, periodicity, etc.). Because output signals are not actually read by humans, response analysis of a system is neither direct nor intuitive. Without suitable visualisation, remote laboratories can be hard to understand for many students. Moreover, interactive RLs should allow the student to simultaneously visualise the response of the real plant to any change introduced by the student. Immediate observation of a change in system response in reaction to user interaction is what really helps the student to develop useful practical insight into control systems theory (Sánchez et al., 2005). Without interactivity, the passivity of students slows down their learning process considerably.

Although the importance of interaction and visualisation is accepted by the engineering education community, their use is not the norm (Phillips, & Rodden, 2001; Uran & Jezernik, 2008). The main reason for this may be that adding interactivity and visualisation to computer applications requires advanced programming skills. Instructors, who are not often programming experts, can run into trouble when trying to add user interaction or advanced visualisation to applications. The variety of different computer languages, programming techniques, network protocols, and so on, makes this task even more complicated.

* Corresponding author.
E-mail address: sebas@dia.uned.es (S. Dormido-Canto).

0360-1315/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compedu.2011.02.015

This paper focuses on providing an approach to teachers for facilitating the creation of remote laboratories for pedagogical purposes. Many advances in the creation of RLs have been done by the control community.

Lazar and Carari (2008) presented an approach to creating networked control systems using the LOOKOUT SCADA software of National Instruments. This laboratory allows the students to develop network-based control systems with server–client applications operating on real pilot plants via the Internet. In this kind of system, the controller and the plant are physically located in different places and are directly linked by a data network for remote closed-loop control. The quality of this control depends greatly on network traffic, which causes delays. However this is a good tool for education in control engineering.

In Dormido et al. (2008), a web-based virtual control laboratory for experimentation with a nonlinear system was presented. Server–client architecture was implemented using Easy Java Simulations (EJS) on the client side and LabVIEW on the server side. The main purpose of this application was for students to learn many fundamental aspects of process control in a practical way. Using this application, students can immediately observe the resulting dynamics and thus become aware of several physical phenomena that are difficult to explain from a purely theoretical point of view. The main difference between this application and our work is that we use MATLAB on the server side.

Gomes and Bogosyan (2009) presented a deep analysis of the current trends in the use of remote laboratories in engineering education and research. The authors described the components, benefits, usage, evolution topologies, platforms, integration with LMS (Learning Management Systems) and similar experiences of remote laboratories. Several of the approaches presented are focused only on remote labs for the simulation of electronic circuits, sequential logic networks, finite-state machine designs, microcomputer interfacing and assembly programming, whereas many applications related to education in automatic control are described only briefly.

The works of Calvo, Zulueta, Oterino, and Lopez-Guede (2009) and Leva and Donida (2008) presented web-based remote laboratories for basic courses in control engineering in which several experiments may be performed using the Ball & Hoop system. Client–server architecture is used here. LabVIEW is used on the server side to acquire and handle process data, whereas OPC technology is used to connect the remote server with the client side. The client side is the remote HMI (human–machine interaction) that students' computers execute within a web browser. These remote applications were programmed with Visual Basic as ActiveX controls that were integrated with Internet Explorer. The Active X controls established a connection to the OPC server available at the server side that provided the process variables of interest to remote users. In parallel with the remote applications, a dedicated video camera was used to provide remote visual feedback to the students. The main limitation of this remote laboratory is that it is not possible to interact with the plant in real time.

Our approach is different to the mentioned alternatives because instructors can use the *de facto* standard software Simulink (The MathWorks Inc., 2009a) as the main tool on the sever side to control the real plant. Simulink is a modelling tool based on MATLAB (The MathWorks Inc., 2009b), which provides a graphical user interface for building models in the form of block diagrams using click-and-drag mouse operations. With this interface, instructors can draw models just as they would with pencil and paper (or as most textbooks depict them). However Simulink diagrams lack of interactivity in the sense that we just described. Instructors also face serious difficulties trying to add interactivity and visualisation to Simulink models. Easy Java Simulations (EJS) is a supplement for this purpose that we use in our approach. EJS is an authoring tool designed to create interactive applications in Java without special programming skills (Esquembre, 2004, 2010).

Using these tools, teachers can build a Simulink diagram to control a real plant and then move to EJS to create graphical user interface with high degree of interactivity and visualisation. A special built-in link of EJS can be used to manipulate the Simulink model (and therefore the real plant) from the interactive user interface. For safety reasons, the real plant is controlled only by one user interface, thus the approach does not allow the manipulation of the remote laboratory by multiple users simultaneously. Although a multi-user scheme similar to the described in (Jara et al., 2009), where one user manipulates the real plant and the rest of users only observe, could be implementing in the future.

This paper is organized as follows. In Section 2, the connection between EJS and Simulink is introduced. Section 3 describes the creation of the remote laboratory using a ball and hoop system. Several experiments with the remote laboratory are shown in Section 4. An education evaluation of the implemented system is discussed in Section 5. Finally, Section 6 presents the main conclusion of the work.

2. Linking EJS and Simulink

In this section, the connection between EJS and Simulink will be described in detail.

2.1. Easy Java Simulations (EJS)

A free software tool for rapid creation of applications in Java, EJS has high level graphic capabilities and an increased degree of interactivity. Applications created by EJS can be stand-alone Java applications or applets; for simplicity, we call these EJS applications or, simply, applications. Source files of EJS applications are saved in a customised xml format. EJS is different from most other authoring tools in that it was designed to make programming easier not for professional programmers but for science students and teachers.

Applications are structured by EJS into two main categories: the model and the view (see Fig. 1). The model can be described by pages of Java code, by ordinary differential equations, or by connections to external applications (such as Simulink). The view provides visualisation of the application and the user-interface elements required for user interaction. View elements can be chosen from a set of predefined components to build a tree-like structure. Models and views can be easily interconnected so that any change in a model state is automatically reflected in the view, and vice versa.

2.2. Using Simulink as an external application in EJS

As previously noted, EJS has a special link for Simulink models. This allows experienced Simulink users to benefit from their expertise to quickly develop interactive applications or to reuse legacy code of existing models.

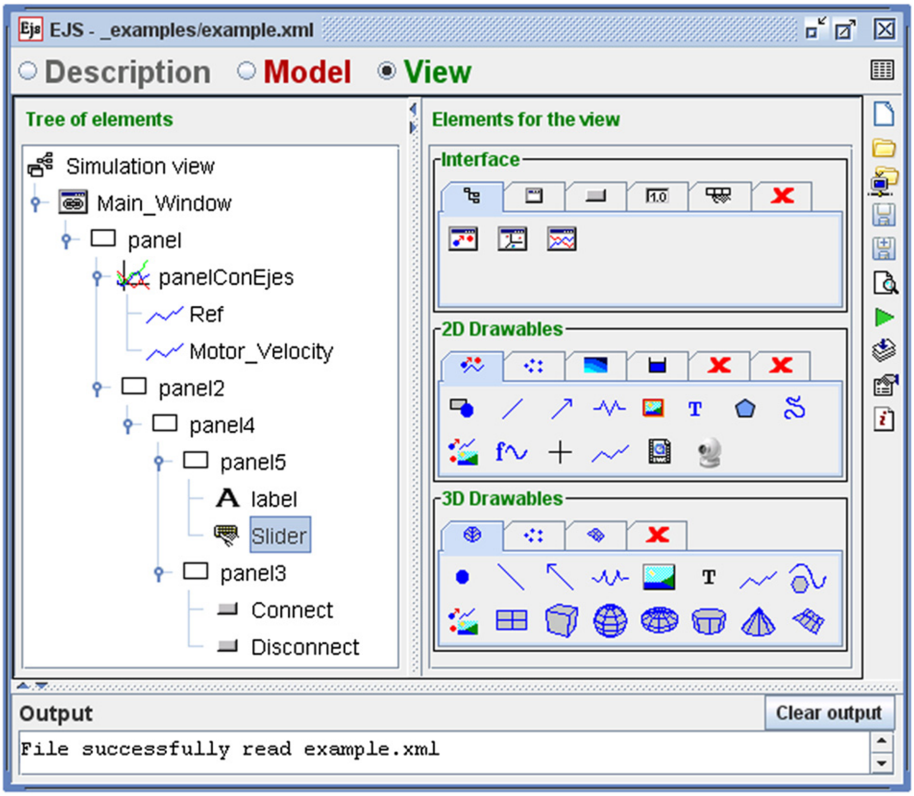

Fig. 1. The view of EJS.

The procedure is quite simple. It consists of connecting EJS variables to the block variables (input, output signals and parameters) of the Simulink model. EJS provides a simple, visual mechanism for this connection, which turns out to be fairly intuitive and which requires no modification of the original Simulink model.

We illustrate this procedure by an example. Consider the Simulink model (*example.mdl*) shown in Fig. 2, which corresponds to an application for interaction with a real plant.

We chose this example for its simplicity and general applicability. As the picture shows, the application allows students to apply a given voltage (using the Step block) to the input of the DC motor and observe velocity in the plot generated by the Scope block. In order to study the influence of a parameter (in this case the motor voltage) on the system's response (in this case the motor velocity), students must execute the Simulink model in three separate steps: 1) set the parameter value (the motor voltage), 2) run the Simulink model, and 3) stop the model and observe system output (the motor velocity plot). Because the output signals are not actually read by a human and the analysis is done off-line, the study is neither explicit nor intuitive; it transforms Simulink models into hard to understand *learning objects* (Gonzalez-Videgaray, Hernandez-Zamora, & Del-Rio-Martinez, 2009).

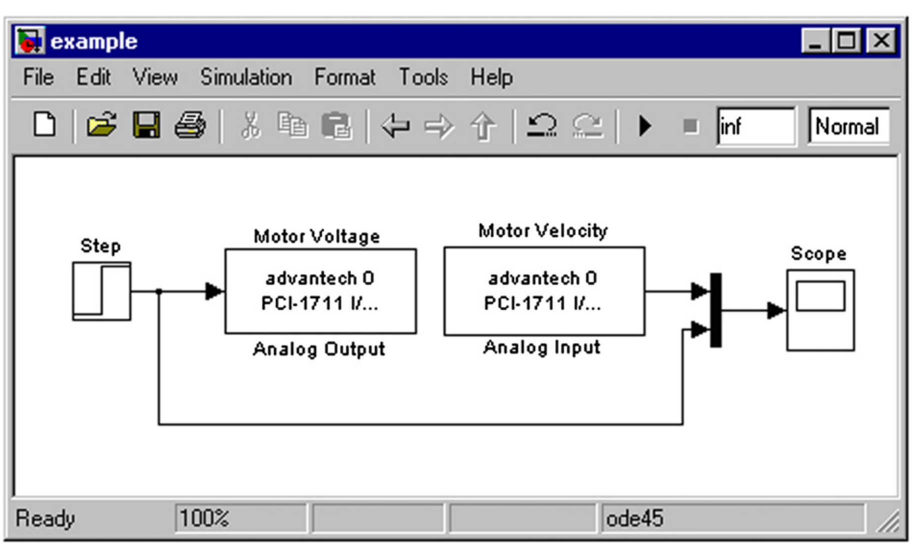

Fig. 2. Simulink model for interaction with a real plant, a DC motor.

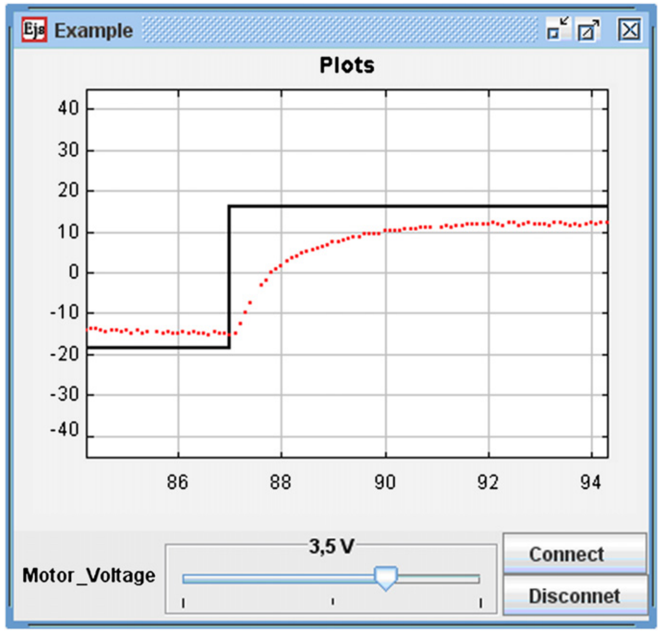

Fig. 3. Graphic user interface of the Java application for interaction with a real plant.

A more suitable learning object comes from using the EJS-Simulink link. Fig. 3 depicts the graphic user interface of the Java application. The application displays a graph plotting the motor voltage (continuous line) and the motor velocity (dotted line). A slider can be used by the student to change the motor voltage interactively; the student can modify the motor voltage while the application is running. There are also two buttons to connect and disconnect the Simulink model. The link between EJS and Simulink is defined by introducing the string "*<matlab>example.mdl*" in the field *External File*, where *example.mdl* represents the Simulink model used.

The *t*, *mv*, and *ref* variables of the application are connected to the Simulink model variables (*time*, *Motor Velocity*, *Motor Voltage*), respectively, as shown in Fig. 4.

Once the variables are connected, the EJS application controls execution of the Simulink model synchronously using a utility method: *_external.step*. (For more details about the connecting process, see Dormido, Esquembre, Farias, & Sánchez, 2005).

2.3. Transforming the local laboratory in RLs

The link between Simulink and the EJS application is direct if both programs run on the same computer. However, if end users do not have MATLAB/Simulink on their computers, they cannot execute the Java application that uses the EJS-Simulink link, and the software JIM server must be used. JIM is a free software tool, written in Java, that allows to an EJS application (or even any Java program) to use a remote MATLAB/Simulink installation in order to execute remotely a Simulink model or a Matlab command and retrieve the outputs (Farias, 2010a, 2010b). There are two kinds of links between JIM and EJS applications: synchronous and asynchronous. The choice between links depends on the type of laboratory that the instructor wants and the context in which the laboratory is used. For RL design, an asynchronous link is useful if the controller is located at the server side, which means that the laboratory is used mainly to monitor the remote plant. Otherwise, if the controller is located in the client side, a synchronous link should be used.

We chose an asynchronous link because the laboratory is used only to monitor the remote plant. To transform the laboratory described in section 2.2 from local to remote, we replaced the string *<matlab>example.mdl* with *<matlabas:10.195.2.57:2005>example.mdl* where *matlabas* sets the asynchronous link and *10.195.2.57:2005* represents the IP address and the port number of the JIM server. Additionally, instructors need to use the built-in function *_external. synchronize()* in order to inform to the remote server user interactions on the RL interface (e.g., by pressing buttons or moving sliders).

The scheme of the connection between EJS applications, JIM and Simulink is shown in Fig. 5. At the client side, there is a Java application created with EJS. The Java application is connected to the JIM server by TCP/IP protocol. At the server side, Simulink is controlled by the JIM

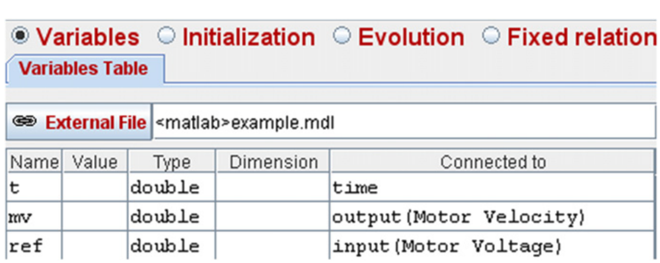

Fig. 4. Connection between variables of application and model.

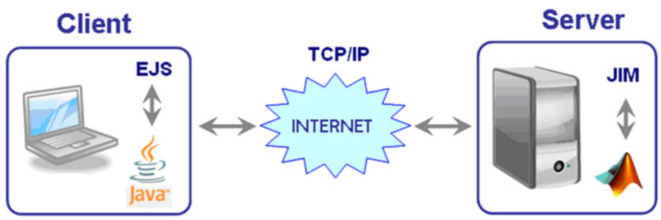

Fig. 5. Scheme of the remote MATLAB-EJS connection.

server to respond to the requests of the Java application. Normally, a Java application that uses this scheme does most of the computation at the server side, whereas the interface is used to show the results and to support the user interaction. Several examples using the JIM server can be found at http://lab.dia.uned.es/rmatlab.

3. Building RLs using a ball-and-hoop system

In this section, the main tools and actions required to create RLs are presented. First, the architecture, the real plant, the software and the hardware are described. Later, the communication processes with the plant are detailed; finally, the client and server side applications are discussed. In our case, we used a ball and hoop system, but the approach can be used with any real plant.

3.1. RL implementation

The laboratory we created is a Java application designed to control a real plant (Ball and Hoop Apparatus CE9) through the Internet. A scheme of the implemented laboratory architecture is presented in Fig. 6. The RLs is divided into three sections: the client side, the network and the server side.

3.2. The real plant (ball and hoop system)

The real plant used for the laboratory is the ball and hoop apparatus CE9 (Wellstead, 1983; Wellstead & Readman, 2000) shown in Fig. 7. The main components of this system are a steel hoop that can rotate about its axis, a steel ball on the hoop's inner periphery, a servomotor that drives the hoop motor torque, a sensor of the hoop angle, a sensor of the hoop velocity, a sensor for the ball angle, and the interface between the plant and the data acquisition card (DAQ Card), which contains some electronic devices for establishing specific operating conditions.

This system is very rich and complex in its dynamics because it has an oscillating but an always-changing behaviour. For this reason, its study is very interesting for our students.

3.3. Software and hardware required

The approach presented in this work is summarised in Fig. 8. As shown, other tools are needed in order to put the control laboratory on-line for students.

In terms of equipment, one needs at least a web server, an IP camera, and a data acquisition card. The web server is required if instructors want to create the remote laboratories as applets. A web server is also needed if the teacher adds virtual laboratories, simulations or documentation to a web site. A good option for this service is Apache Tomcat Software, although any web server such as Internet Information Server can be used in this approach. The Apache Tomcat server also offers security functions (as normally all popular web servers). These security features of Apache Tomcat (called *Authorization*, *Authentication* and *Access control)* allow access only to authorised students. A simple web camera can be used to show students a view of the real plant. However, it is preferable to use an IP camera because it has a built-in web server that can stream video images directly to the Internet. The EJS application has a specific visual element to display video from stream servers, so access to the streams of IP cameras is quite direct and simple to use in EJS.

There are many options when developers need to control external hardware (plants) from computers. In our case different data acquisition (DAQ) cards can be used; the only restriction in this approach is that the selected card has to be compatible with MATLAB. Communication with the DAQ card can be established using the data acquisition toolbox of MATLAB.

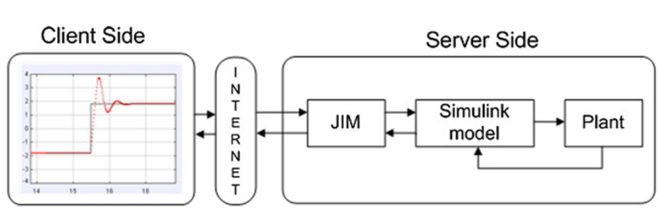

Fig. 6. Architecture of remote laboratories.

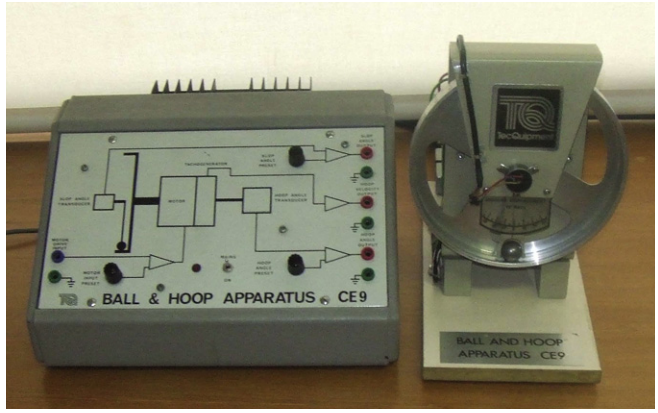

Fig. 7. Ball and Hoop Apparatus CE9.

3.4. Acquisition process from DAQ cards

MATLAB's data acquisition toolbox provides a complete set of tools for analog input, analog output, and digital I/O from a variety of PC-compatible data acquisition boards. The toolbox lets us configure external hardware devices, read data into the MATLAB environment for immediate analysis, and send out data. The data acquisition toolbox also supports Simulink with blocks that enable us to incorporate live data or hardware configuration directly into Simulink models. The toolbox provides users with these main features (The MathWorks Inc, 2010):

- Acquires live, measured data directly into MATLAB or Simulink for immediate analysis.
- Provides a single integrated environment for data acquisition, analysis, and visualisation.
- Performs "one-shot" or continuous data acquisition.
- Configures and accesses analog input, analog output, and digital I/O.
- Controls acquisitions with hardware and software triggers.
- Provides a consistent software interface for easy substitution of hardware boards and vendors.

We use the Advantech PCI-1711L data acquisition card. Fig. 9 shows the Simulink blocks of the data acquisition toolbox available for this device.

3.5. Client-side application

The application on the client side is a Java application created using EJS. The main window is divided into three parts, as shown in Fig. 10. The upper part shows the video signal of the IP camera through which the student can see what happens at the real plant located at the university laboratory. There is a set of buttons in the middle of the main window for managing experimental operation (Play, Stop and Pause). The bottom panel of the main window has a tab for modifying the parameters of the PID controller, depending on the experiment.

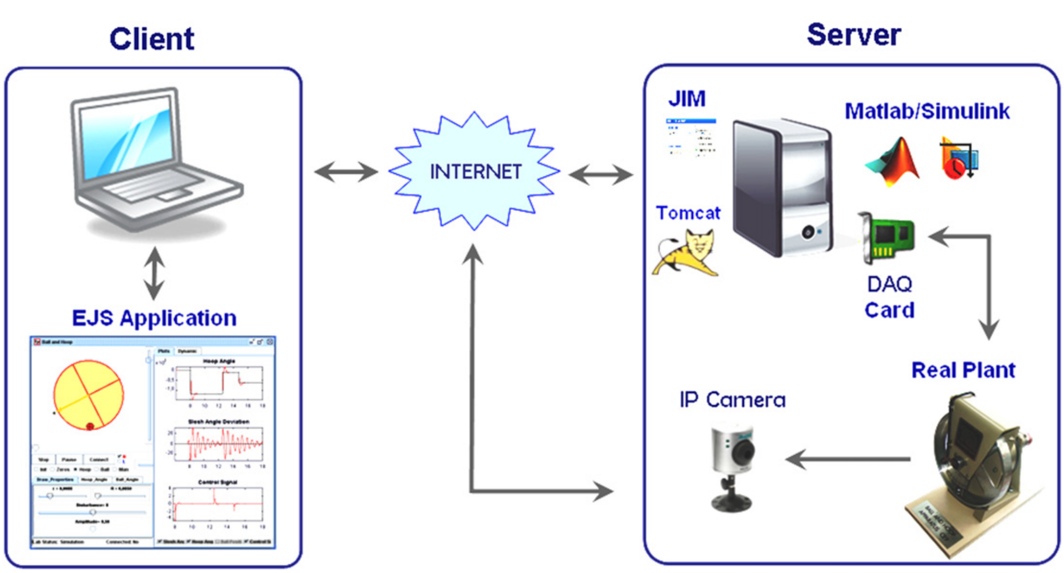

Fig. 8. Elements required for an RL implemented with the Simulink-EJS approach.

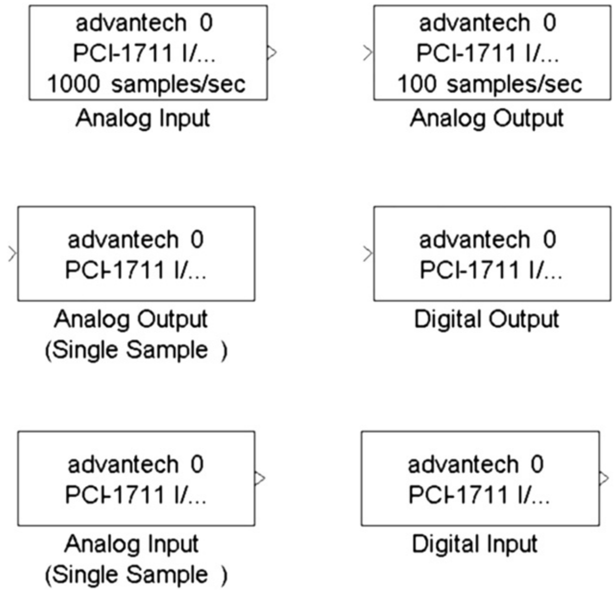

Fig. 9. Simulink blocks of the data acquisition toolbox for DAQ card Advantech PCI-1711L.

Students can use sliders to manipulate variables like Motor Voltage. There is a tabbed panel with two tabs at the right side of the main window. The first tab, titled Plots, contains the graphics representations of the hoop angle, the ball angle and the control signal of the real plant with respect to the time. The student can select which plotting panel to set visible or invisible, depending on its utility for the experiment. The other tab of the panel is titled Dynamic; it contains an interactive root-locus representation of the system.

3.6. Server side

The architecture of the RL is based on a single client–server structure; that is, the same computer functions as both the server and the controller for the physical plant. The server application was developed using Simulink (see Fig. 11) as a stand-alone application and can be manipulated by the JIM server when the student connects to the RL. We can obtain the real plant values (hoop angle, ball angle and hoop velocity) by using the blocks from the data acquisition toolbox of Simulink. At the same time, we can send a voltage to the input of the motor. The main control loop is a typical feedback position control loop for a DC motor and has an internal feedback of hoop velocity for better stability. The secondary control loop is for feedback of the ball angle. The controller used is a parallel PID.

4. Experiments with RLs

In this section, results from RLs for the ball and hoop are presented.

4.1. Hoop position control

The experiment of hoop angle control consists of leading the position of the hoop to a given reference. A suitable set of the controller parameters must be obtained in order to get a good system response. The control loop of this experiment is shown in Fig. 11, with the gain K of the ball angle equal to zero. In this condition, when a step is applied to the input of the system, the hoop angle follows the reference; the ball oscillates until it reaches its rest position. The reference can be changed by dragging and dropping directly from the slider situated at the bottom of the video (see Fig. 10). An edit field shows the current value of the hoop angle reference in degrees. The controller parameters can be changed in the tab titled Hoop_Angle. This experiment is the basis for the next one; both are shown in Fig. 10.

4.2. Ball angle deviation

To demonstrate the deviation angle of the ball, we need to have the hoop angle under feedback control and use the best set of PID controller parameters. This experiment consists of change the value of the gain K (which was zero for the previous experiment) to move the hoop and at the same time and cause minimal displacement of the ball. When increasing this value, the ball angle is taken into account. Then we can observe how, for a step applied to the reference, the hoop responds more slowly, causing a minimal deviation of the ball's equilibrium point. For small values of K, the ball is only lightly damped, while the hoop response is fast. For large values of K, the hoop's position is sluggish. Fig. 10 (from the time value 291) shows the behaviour of the system as K changes.

Other experiments that can be demonstrated with this real plant are the zeros of transmission, which consists of applying a sine wave to the hoop angle at the frequency of the exactly-zero response, and non-minimal phase behaviour, which is observed by subtracting the ball angle from the hoop angle and plotting the result. (More details about experiments using the ball and hoop system can be found in Wellstead & Readman, 2000).

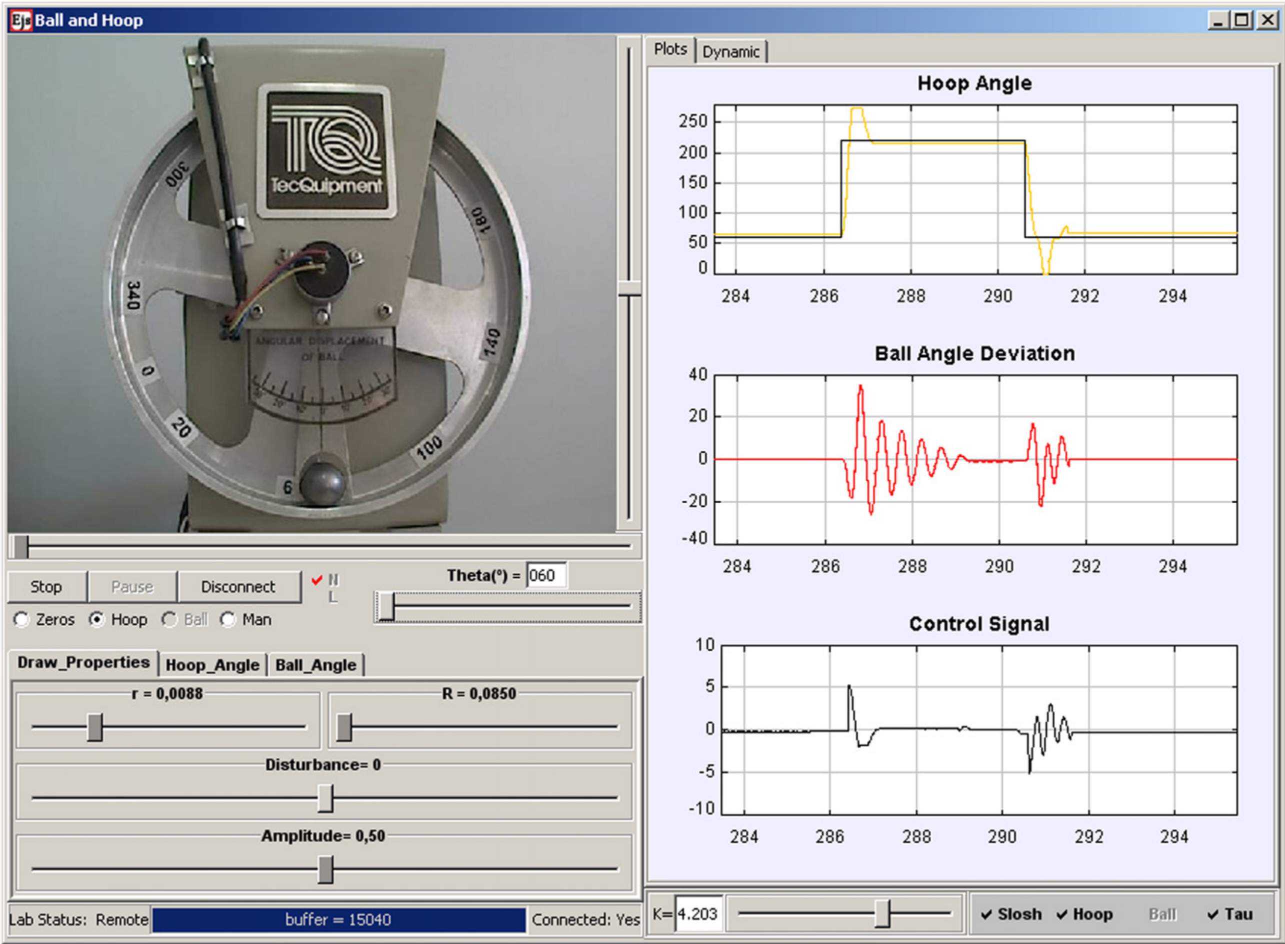

Fig. 10. Main window of the client application.

5. Educational evaluation

Laboratory practices have been used for several years for courses in automatic control in the Computer Science Engineering degree at the National University of Distance Education in Spain. In previous years, students had to perform work by themselves. The work was to study the system and analyse the theory of the experiments that would be performed. Then they were to attend the laboratory in the University, to meet face to face with the real plant, conduct experiments and compare their results with previous work. In this way, students came to know the real problem that they will face. At the end of the process, students submitted a report containing the results and conclusions obtained. This report was evaluated by the professor. The results and the final test score determined whether students passed the course. In the current course, laboratory practices have been improved following the principles of traditional laboratory practices (homework tasks in addition to laboratory practice). The principal modifications were adding remote experimentation with a real plant as complementary activity to the traditional laboratory. It is clear that the use of RL prior to the real lab should benefit the students, because they will better and

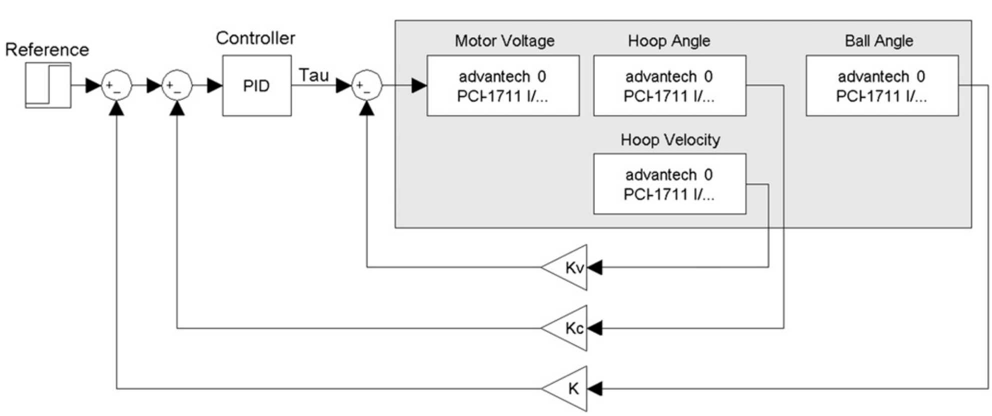

Fig. 11. Simulink model for the ball and hoop (server application).

more quickly understand the behaviour of the real equipment if they have previously used the RL. This situation reduces the cognitive load required (van Merriënboer & Sweller, 2005) to acquire new knowledge of control theory.

The educational methodology of current laboratories can be defined as follows:

- Homework tasks: In this stage, students are required to complete experimentation through the Internet (pre-lab assignments) with the RL of the ball and hoop system plant. The virtual laboratory developed in (Fabregas, Duro, Dormido, Dormido-Canto, Vargas, & Dormido, 2009) provides the same graphic user interface of the RLs (see Fig. 10) but with simulated data. The main activities required of the students are related to identification and controlling (design and tuning) tasks of the simulated or real plant.
- Experimentation at the university: After working at home with the RL, students must attend the face to face laboratory, run an extended version of the control experiments described in the pre-lab assignments, and check the previous results. Finally, students have to choose one of the additional available experiments with the ball and hoop, such as the zeros of transmission, among others (Wellstead & Readman, 2000).

At the end of the experimentation stages, students were invited to evaluate the RL using an on-line poll (which allowed students to vote confidentially). It was explained to the students that the objective of the evaluation of the laboratory was to get their views on the contribution of remote experimentation to their development as engineers. Questionnaire items, which were based on the work of (Dormido et al., 2008; Jara et al., 2009), were combined in four subscales:

- *Learning Value* includes items that reflect students' perceptions of how effectively the laboratory (virtual-remote-practical) helps them learn the relevant contents.
- *Value Added* by the laboratory with remote experimentation is assessed, along with its advantages over traditional laboratories.
- *Design Usability* of the laboratory focuses on students' perceptions of the ease and clarity with which they are able to navigate through the laboratory.
- *Technology Function* assesses students' perceptions of how well the laboratory functioned technically and whether they had the technical knowledge required to use it.

Table 1 shows the questions of the on-line poll presented to a group of 30 students who were randomly selected.

The answers to each item were averaged for each subscale and rated as strongly agree, agree, neutral, disagree, or strongly disagree (see an example of the use in Gurocak, 2001). Table 2 shows the results as percentages for each subscale.

The results indicate that about 69% of students think that the laboratory with remote experimentation helped them to understand relevant concepts (Learning Value), whereas only 16% disagreed or strongly disagreed. Regarding Value Added, 59% of the students found that the laboratory with remote experimentation has advantages over the traditional labs; however about 23% of the students found the opposite. This item may indicate the value that students give to the traditional practices in the laboratory, which means that the remote or virtual activities should be considered as a complement (and not a replacement) in control engineering teaching. Regarding Design Usability, about 62% of students had no difficulty using the laboratory. Technology Function was not a problem for most of students, as only 13% evaluated this category as "disagree" or "strongly disagree".

The general criteria of the students who used the laboratory can be summarised as follows:

- More time to interact with the system through remote access to the real plant allows them a greater understanding of physical phenomena that may occur.

Table 1
On-line questionnaire used to obtain user feedback.

Learning Value
- Did the lab enhance your ability to understand the theoretical material in a new way? - Did the lab help you to visualise the concept to be learned? - Did you gain as much information as you would from a lecture explanation?
Value Added
- Were you able to develop a better understanding of how to control these kinds of systems? - Were you able to work in a way that would not have been possible by attending a traditional lab? - Was the level of interactivity in the laboratory adequate? –Do you think that time face to face with the real plant was enough? - Does the remote access and virtual experiment allow you to be better prepared for the lab?
Design Usability
- Was the laboratory easy to understand and use? - The ideas and concepts incorporated within the laboratory were clearly presented and easy to follow? - Were you able to fully use the laboratory by following the instructions provided? - Were laboratory handouts useful?
Technology Function
- Did you miss important information because the technology did not work correctly? - Did the software requirements pose a problem for you? - How was the response time of the laboratory?

Table 2
Student questionnaire results in percentage of agreement per subscale.

Subscale	Strongly agree (%)	Agree (%)	Neutral (%)	Disagree (%)	Strongly disagree (%)
Learning Value	30	40	13	13	3
Value Added	23	37	17	17	7
Design Usability	20	43	27	3	7
Technology Function	17	57	13	10	3

- The RL is an efficient tool in the learning process that encourages their ability to understand control concepts.
- The availability of these tools requires more commitment and time to perform the experiments.
- Remote experimentation allows a better analysis of the theoretical and practical arguments.

Suggestion: simulation should be done in groups to share criteria. The teacher should be on-line during remote experimentation in case any help is needed.

To add to previous results about the evaluation of remote experimentation in the laboratories practices, a new study was carried out to verify its usefulness and efficiency as complement to traditional laboratories. The study was an analysis of the impact of remote experimentation on the academic performance of students and its influence on the quality of the learning process. The academic results, reports of laboratory practice and final exams, of 30 students (from a total of 43 of the previous year, when RL was not used) and 30 students (from a total of 42 of this course, which used the RL) were analysed in detail for comparison. Fig. 12 shows marks of both groups of students in the report of the laboratory practice and in the final exam.

On the left side is shown the results of the reports on lab practice. Both groups of students who used a traditional lab system (Classic Lab) and students who used the new proposed system (Remote Lab) resolved experiments with similar marks. However, a few students got better marks after using remote experimentation: 6.66% (2 students) more scored an A qualification and 13.33% (4 students) more scored a B qualification. The general quality of the laboratory report of students who used the developed tool is a noticeable improvement mainly in the statements of the conclusions and in the explanations of physical phenomena (similar to the results described in Ogot, Elliott, & Glumac, 2003). This is due to a better analysis of argument and from linking theory with practice, because students can spend more time with the real plant and therefore better analyse and understand the physical phenomena that take place. In the right side of Fig. 12 are shown the results of the final exams for both groups of students. All students passed the final exam. As in the previous case, a few students got better marks after using the remote experimentation: 13.33% (4 students) more had an A qualification, and 10% (3 students) more had a B qualification. The difference in the final exam between the two years increased, indicating that students were better prepared. In general, the learning of relevant concepts was improved by the students who used the RL when compared to other students (similar to the study shown in Henson, Fridley, Pollock, & Brahler, 2002). These results demonstrate the positive influence of remote experimentation in the development of engineering students, in the acquisition of a new practical view of their theoretical knowledge as described in Alexander and Smelser (2003) and Madhavan, Schroeder, and Xian (2009).

The analysis of the influence of the experiments in the academic performance of students has the following advantages:

- The academic results of students who used the remote laboratory can be compared with those who did not use it.
- The study is based in the marks obtained by the students. Personal criteria for marks do not influence the results of the study.
- The quantitative analysis of these results can show whether the learning of relevant concepts is improved by using the remote laboratory.

The main limitations of this analysis are:

- The final exams were not the same but the relevant concepts evaluated in both are very similar.
- The reliability of the data is only based on the random selection of the students for the analysis.
- In order to evaluate the quality of the results, the laboratory report conclusions must also be analysed.
- The study was conducted without a statistical analysis of the results. It would have been better to perform an analysis of variance (ANOVA) on the scores for the entire group of students selected, as in Henson et al. (2002), Rutz et al. (2003) and Gurocak (2001).

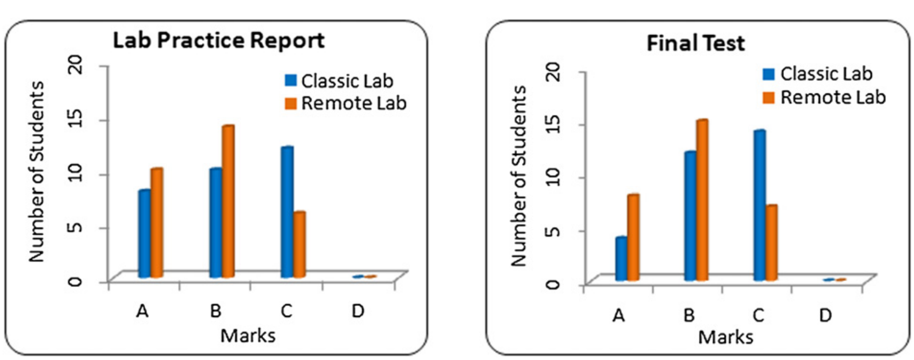

Fig. 12. Results using a Classic Lab vs. a Remote Lab.

A suggestion to improve the study is to use another methodology to select the students for comparison. For example, apply an initial test and divide the students into groups by score as follows. The students are separated into pairs with similar scores. Then, two groups (A and B) are selected from the pairs obtained, where group A is composed of the first member of each pair and group B of the second member of each pair. In this way, you can ensure that the groups have the same average scores at the beginning of the study. Then, perform the study in a way similar to what was described in this work. With this methodology, the groups selected for comparison will be more homogeneous (this suggestion is presented in Henson et al., 2002).

6. Conclusions and further work

In this paper, an approach to build interactive remote laboratories has been presented. The work allows non-programming instructors to create innovative pedagogical tools that can be used to motivate students to apply the theory of automatic control to new challenges like the remote control of plants. This kind of laboratory can be used as part of a basic engineering control course as a complement of the traditional laboratories. This practical experimentation is very important for engineering students in their development as engineers. In this sense, RLs allows them to have a second chance to face with real plants to better study and understand the objectives of the analysed process. For this reason, the RLs are a complement to traditional laboratories, not a substitute. In last years, RLs has been gaining ground to the traditional laboratories, so it is possible that in the future, with the development of information technologies and communications, the traditional labs can be replaced by the RLs.

Two main software tools, Simulink and Easy Java Simulations, were used to implement the approach. Simulink was selected because it is a very well known tool in the control engineering community, whereas Easy Java Simulations allows teachers to build complex Java applications with high levels of interactivity and visualisation but with minimum skills in computer programming. To illustrate the use of the approach, the implementation of RLs with a ball and hoop apparatus was described. The ball and hoop system was selected because of its special dynamic characteristics that make it an excellent tool for demonstrating aspects of control theory in engineering education. A virtual laboratory of a ball and hoop system was part of an Automatic Control course at the University. The educational methodology was divided into two stages. First, students worked at home with RLs. Second, students faced a real plant at the university following assignments similar to those required in pre-lab activities. The results of this methodology showed that students improved their understanding of the theoretical and practical knowledge. From students' opinions, it can be concluded that students appreciated pre-lab assignments with the RLs; however, it also can be inferred that this kind of learning resources cannot be used as a replacement but as a complement of traditional laboratories.

A study was carried out in order to evaluate the laboratory from the pedagogical point of view. A group of students of the current course were invited to participate in the study with its criteria. The results indicated that the students think that the laboratory with remote experimentation helped them to understand relevant concepts. The main criterion of the students who used the laboratory was that they have more time to interact with the system through the remote access to the real plant. They found the RL to be an efficient tool for understanding relevant control concepts and physical phenomena that may occur. In addition, experimentation with real plant awakens their curiosity and motivation to learn.

An analysis of the academic results of the students was carried out to complement the study and evaluate the influence of the RLs. In this case, the students were separated in two groups for comparison (those who did and those who did not use RLs). For the laboratory report a few students who got better marks with the use of the RLs. The general quality of the laboratory report of students who used the tool was noticeably improved, mainly in the statements of the conclusions and in the explanations of physical phenomena. For the final exam, a few students got better marks after using RLs. The marks on the final exam increased, indicating that students were better prepared. In general, the learning of relevant concepts was improved by the students who used the RLs as compared to the other students. These results demonstrate the positive influence of the remote experimentation in its development as engineering students. In the future, to obtain better results in the study, we would perform a statistical analysis of variance (ANOVA) on the scores of the selected students.

Regarding future works the main focus could be related with the use of a Web infrastructure to support the learning process of students in a distributed scenario. This platform should organize user access to the experimentation modules that are available and allow for students/teachers to interact and collaborate with one another.

Automatic assessment could also be an interesting matter for future research. Normally students are asked to answer quizzes or do exercises during the use of the virtual and remote labs in order to ensure the understanding of critical knowledge. However, the more quizzes are asked, the more time to evaluate the quizzes is required. Hence, the use of automatic systems to help teachers in the evaluation of the RL process could also be an interesting topic for future research.

Another topic that could be taken into account in the future is to run experiments with existing virtual labs programmatically. This should allow students to execute several simulations with different initial conditions at the same time, which provides a better understanding of the effect of variables in the system's response.

Finally, it would be very interesting for future research to add a network wherein the control loop. This alternative architecture puts the controller at the client side, which is different to the widely used traditional arquitecture that has the controller at the server side. The insertion of the communication network in the feedback control loop makes the analysis and design of these kind of systems complex and challenging. Network delays in control loops can impose severe degradation on system performance. These delays have interesting consequences from the control education point of view.

References

Alexander, D. G., & Smelser, R. E. (2003). Delivering an engineering laboratory course using the Internet, the post office, and a campus visit. *Journal of Engineering Education, 92*(1), 79–84.

Calvo, I., Zulueta, E., Oterino, F., & Lopez-Guede, J. M. (2009). A remote laboratory for a basic course on control engineering. *International Journal of Online Engineering, 5*(3).

Dormido, R., Vargas, H., Duro, N., Sánchez, J., & Dormido-Canto, S. (2008). Development of a web-based control laboratory for Automation Technicians: the three-Tank system. *IEEE Transactions on Education, 51*(1), 35–44.

Dormido, S. (2004). Control learning: Present and future. *IFAC Annual Reviews in Control, 28*, 115–136.

Dormido, S., Esquembre, F., Farias, G., & Sánchez, J. (2005). Adding interactivity to existing Simulink models using Easy Java Simulations. *44th IEEE Conference on Decision and Control and European Control Conference ECC*, Seville, Spain.

Esquembre, F. (2004). Easy Java Simulations: a software tool to create scientific simulations in Java. *Computer Physics Communications*, *156*, 199–204.

Esquembre, F. (2010). *Easy Java simulations home Page*. http://fem.um.es/Ejs.

Fabregas, E., Duro, N., Dormido, R., Dormido-Canto, S., Vargas, H., & Dormido, S. (2009). Virtual and remote experimentation with the Ball and Hoop system. *14th International Conference on Emerging Technology and Factory Automation*, Palma de Mallorca, Spain.

Farias, G. (2010a). *Adding Interactive Human Interface to Engineering Software*. PhD Thesis. UNED. Spain.

Farias, G. (2010b). *JIM server home Page*. http://lab.dia.uned.es/rmatlab/.

Gomes, L., Coito, F., Costa, A., Palma, L., & Almeida, P. (2007). Remote Laboratories support within Teaching and Learning Activities. *International Conference on Remote Engineering and Virtual Instrumentation*.

Gomes, L., & Bogosyan, S. (2008). *Editorial IEEE Transactions on Industrial Electronic, 55*(6), 2324–2325.

Gomes, L., & Bogosyan, S. (2009). Current trends in remote laboratories. *IEEE Transactions on Industrial Electronics, 56*(12), 4744–4756.

Gonzalez-Videgaray, M., Hernandez-Zamora, G., & Del-Rio-Martinez, J. (2009). Learning objects in theory and practice: a vision from Mexican University. *Computers & Education, 53*(4), 1330–1338.

Gurocak, H. (2001). e-Lab: an electronic classroom for real-time distance delivery of a laboratory course. *Journal of Engineering Education, 90*(4), 695–705.

Heck, B. S. (Ed.). (1999), *Special report: future directions in control education, vol. 19* (pp. 35–58). IEEE Control System Magazine, 5.

Henson, A. B., Fridley, K. J., Pollock, D. G., & Brahler, C. J. (2002). Efficacy of interactive Internet-based education in structural timber design. *Journal of Engineering Education, 91*(4), 371–378.

Jara, C., Candelas, F. A., Torres, F., Dormido, S., Esquembre, F., & Reinoso, O. (2009). Real-time collaboration of virtual laboratories through the Internet. *Computers & Education, 52*(1), 126–140.

Lazar, C., & Carari, S. (2008). A remote-control engineering laboratory. *IEEE Transaction on Industrial Electronics, 55*(6), 2368–2375.

Leva, A., & Donida, F. (2008). Multifunctional remote laboratory for education in automatic control: the crautolab experience. *IEEE Transactions on Industrial Electronics, 55*(6), 2376–2385.

Madhavan, K., Schroeder, J., & Xian, H. (2009). *Evaluating the Effectiveness and use of Cyberlearning environments in engineering education—A Qualitative analysis*. American Society for Engineering Education.

Ogot, M., Elliott, G., & Glumac, N. (2003). An assessment of in-person and remotely operated laboratories. *Journal of Engineering Education, 92*(1), 57–64.

Phillips, P., & Rodden, T. (2001). Multi-authoring virtual worlds via the World Wide Web. *Interacting with Computers, 13*(3), 401–426.

Rutz, E., Eckar, R., Wade, J. E., Maltbie, C., Rafter, C., & Elkins, V. (2003). Student performance and acceptance of instructional technology: Comparing technology-Enhanced and traditional Instruction for a course in Statics. *Journal of Engineering Education, 92*(2), 133.

Sánchez, J., Dormido, S., & Esquembre, F. (2005). The learning of control concepts using interactive tools. *Computer Applications Engineering Education, 13*(1), 84–98.

The MathWorks, Inc. (2009a). *The MathWorks I Simulink User's Guide*.

The MathWorks, Inc. (2009b). *The MathWorks I MATLAB Getting Started Guide*.

The MathWorks, Inc. (2010). *Data acquisition toolbox 2.15*. The MathWorks home Page. http://www.mathworks.es/products/daq/.

Uran, S., & Jezernik, K. (2008). Virtual laboratory for Creative control design experiments. *IEEE Transactions on Education, 51*(1), 69–75.

van Merriënboer, J. G., & Sweller, J. (2005). Cognitive load theory and complex learning: Recent developments and future directions. *Educational Psychology Review, 17*(2).

Wellstead, P. E. (1983). The ball and hoop system. *Automatica, 19*(4), 401–406.

Wellstead, P. E., & Readman, M. (2000). *Ball and Hoop: Control and analysis*.